CLAIMS

1. A method of tracking data through a multi-tier computing architecture, the method comprising:

initializing a first row pointer of a dataset with data from a database;

placing changes to the data in a second row pointer of the dataset;

comparing the first and second row pointers; and

declaring a state of the data in response to comparing the first and second row pointers.

2. A method according to claim 1, wherein:

initializing includes initializing an original row pointer of a dataset with data from a database.

3. A method according to claim 2, wherein:

placing includes placing changes to the data in a current row pointer of the dataset.

4. A method according to claim 3, wherein:

comparing includes comparing the original row pointer to the current row pointer to determine the state of the data.

5. A method according to claim 1, further comprising:

receiving a request from a client computing system for data from a database, wherein:

initializing includes initializing a first row pointer of a dataset with the

data requested by the client computing system in response to the request by the client computing system.

6. A method according to claim 1, further comprising:

sending the initialized dataset to the client computing system; wherein

placing changes includes placing changes to the data by the client computing system in a second row pointer of the dataset; and

receiving the dataset with the changes.

7. A method according to claim 1, wherein:

comparing includes detecting whether the first row pointer is null, and if the first row pointer is null, declaring the data to be new data in response to detecting the first row pointer to be null.

8. A method according to claim 1, wherein:

comparing includes detecting whether the second row pointer is null, and if the second row pointer is null, declaring the data to be deleted data in response to detecting the second row pointer to be null.

9. A method according to claim 1, wherein:

comparing includes detecting whether the first and second row pointers are equal, and if the first and second row pointers are equal, declaring the data to be original data in response to detecting the first and second row pointer to be equal.

10. A method according to claim 1, wherein:

comparing includes detecting whether the first and second row pointers are equal, and if the first and second row pointers are not equal, declaring the data to be updated data in response to detecting the first and second row pointers to not be equal.

11. A method according to claim 1, further comprising:

committing the data in the second row pointer.

12. A method according to claim 11, wherein:

committing includes accepting, rejecting, or merging the data.

13. A method according to claim 12, wherein:

merging includes merging a plurality of datasets from a plurality of client computing systems.

14. A method according to claim 13, wherein:

merging includes matching locally unique identifiers between rows of the plurality of datasets.

15. A method according to claim 11, wherein:

committing includes updating the database with the data in the second row pointer.

16. A computer program product readable by a computing system and encoding instructions for a computer process for tracking data through a multi-tier computing

architecture, the computer process comprising:

initializing a first row pointer of a dataset with data from a database;

placing changes to the data in a second row pointer of the dataset;

comparing the first and second row pointers; and

declaring a state of the data in response to comparing the first and second row pointers.

17. A computer program product according to claim 16, wherein:

initializing includes initializing an original row pointer of a dataset with data from a database.

18. A computer program product according to claim 17 wherein:

placing includes placing changes to the data in a current row pointer of the dataset.

19. A computer program product according to claim 18, wherein:

comparing includes comparing the original row pointer to the current row pointer to determine the state of the data.

20. A computer program product according to claim 16, further comprising:

receiving a request from a client computing system for data from a database, wherein:

initializing includes initializing a first row pointer of a dataset with the data requested by the client computing system in response to the request by the client computing system.

21. A computer program product according to claim 16, further comprising:

sending the initialized dataset to the client computing system; wherein

placing changes includes placing changes to the data by the client computing system in a second row pointer of the dataset; and

receiving the dataset with the changes.

22. A computer program product according to claim 16, wherein:

comparing includes detecting whether the first row pointer is null, and if the first row pointer is null, declaring the data to be new data in response to detecting the first row pointer to be null.

23. A computer program product according to claim 16, wherein:

comparing includes detecting whether the second row pointer is null, and if the second row pointer is null, declaring the data to be deleted data in response to detecting the second row pointer to be null.

24. A computer program product according to claim 16, wherein:

comparing includes detecting whether the first and second row pointers are equal, and if the first and second row pointers are equal, declaring the data to be original data in response to detecting the first and second row pointers to be equal.

25. A computer program product according to claim 16, wherein:

comparing includes detecting whether the first and second row pointers are equal, and if the first and second row pointers are not equal, declaring the data to

be updated data in response to detecting the first and second row pointers to not be equal.

26. A computer program product according to claim 16, further comprising:

committing the data in the second row pointer.

27. A computer program product according to claim 26, wherein:

committing includes accepting, rejecting, or merging the data.

28. A computer program product according to claim 27, wherein:

merging includes merging a plurality of datasets from a plurality of client computing systems.

29. A computer program product according to claim 28, wherein:

merging includes matching locally unique identifiers between rows of the plurality of datasets.

30. A computer program product according to claim 26, wherein:

committing includes updating the database with the data in the second row pointer.

31. A system for tracking data through a multi-tier architecture, the system comprising:

an initialize module that initializes a first row pointer of a dataset with data from a database;

a change module that places changes to the data in a second row pointer of the

dataset;

a compare module that compares the first and second row pointers; and

a declare module that declares a state of the data in response to comparing the first and second row pointers.

32. A system according to claim 31, wherein:

the first row pointer is the original data pointer; and

the second row pointer is the current row pointer.

33. A system according to claim 31, further comprising:

a receive module that receives a request from a client computing system for data from a database, wherein:

the initialize module initializes the first row pointer of the dataset with the data requested by the client computing system in response to the request by the client computing system.

34. A system according to claim 31, further comprising:

a send module that sends the initialized dataset to the client computing system; wherein:

the change module places changes to the data by the client computing system in the second row pointer of the dataset, and receives the dataset with the changes.

35. A system according to claim 31, further comprising:

a first null module that detects whether the first row pointer is null; and

a new module that declares the data to be new data in response to detecting the first row pointer to be null.

36. A system according to claim 31, further comprising:

a second null module that detects whether the second row pointer is null; and a delete module that declares the data to be deleted data in response to detecting the second row pointer to be null.

37. A system according to claim 31, further comprising:

an equal module that detects whether the first and second row pointers are equal;

an original module that declares the data to be original data in response to detecting the first and second row pointers to be equal; and

an updated module that declares the data to be updated data in response to detecting the first and second row pointers to not be equal.

38. A system according to claim 31, further comprising:

a commit module that commits the data in the second row pointer.

39. A system according to claim 38, wherein:

the commit module includes an accept module, a reject module, and a merge module.

40. A system according to claim 39, wherein:

the merge module merges a plurality of datasets from a plurality of client computing systems.

41. A system according to claim 40, wherein:

the merge module matches locally unique identifiers between rows of the plurality of datasets.

42. A system according to claim 38, wherein:

the commit module updates the database with the data in the second row pointer.